

Appl. No. 10/056,880
Amdt. Dated July 1, 2004
Reply to Office Action of April 9, 2004

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Original) A method of performing a join in a database system comprising:
receiving a join query containing at least one function selected from the group consisting of a selection predicate applied on a complex attribute, a projection applied on a complex attribute, and a user-defined data type method;
determining a cost associated with applying the function on a first table and a cost associated with applying the function on a second table; and
selecting a join path based on relative costs of applying the function on the first and second tables.
2. (Original) The method of claim 1, wherein selecting the join path comprises applying the function on one of the first and second tables associated with a lower cost.
3. (Original) The method of claim 1, wherein determining the costs comprises determining the respective cardinalities of the first and second tables.
4. (Original) The method of claim 3, wherein determining the cost of applying the function on the second table comprises determining the cost of a join table that is a result of a join of the first table and another table.
5. (Original) The method of claim 3, wherein selecting the join path comprises applying the function on one of the first and second tables that has the lower cardinality.
6. (Original) The method of claim 5, wherein the function comprises a selection predicate applied on a complex attribute of the first table, the join query further containing a projection applied on a complex attribute of the first table, the method further comprising:
determining a cost associated with applying the projection on the first table and a cost associated with applying the projection on the join table,

Appl. No. 10/056,880
Amdt. Dated July 1, 2004
Reply to Office Action of April 9, 2004

wherein selecting the join path comprises applying the projection on one of the first table and the join table associated with a lower cost.

7. (Original) The method of claim 6, wherein selecting the join path comprises applying the projection on one of the first table and join table with the lower cardinality.

8. (Original) The method of claim 1, further comprising identifying the function as a costly function.

9. (Original) The method of claim 1, wherein the receiving, determining, and selecting acts are performed by an optimizer module.

10. (Original) The method of claim 1, wherein determining the costs of applying the function on the first and second tables comprises determining the costs of applying the function on object relational tables.

11. (Currently Amended) An article comprising at least one storage medium containing instructions that when executed cause a database system to:

receive a join query containing at least one function selected from the group consisting of a selection predicate applied on a complex attribute, a projection applied on a complex attribute, and a user-defined data type method, the join query specifying a join of a first table and a second table to produce a join table; and

determine a join path for the join query based at least in part on a cost associated with application of the function ~~on the complex attribute,~~

wherein determining the join path comprises selecting the join path in which the function is applied on the join table rather than the first table or second table to reduce cost.

12. (Currently Amended) The article of claim 11, wherein the join query specifies the function being applied on ~~[[a]]~~ the first table, and wherein the instructions when executed cause the database system to determine the join path in which by applying the function is applied on the join ~~a second table different from the first table.~~

13. (Canceled)

Appl. No. 10/056,880
Amdt. Dated July 1, 2004
Reply to Office Action of April 9, 2004

14. (Currently Amended) ~~The article of claim 11,~~ An article comprising at least one storage medium containing instructions that when executed cause a database system to:
receive a join query containing at least one function selected from the group consisting of a selection predicate applied on a complex attribute, a projection applied on a complex attribute, and a user-defined data type method;

determine a join path for the join query based at least in part on a cost associated with application of the function,

wherein the join query specifies the function being applied on a first table[.,,];

and

~~wherein the instructions when executed cause the database system to determine the join path by applying the function on a second table having a lower cardinality than the first table.~~

15. (Canceled)

16. (Currently Amended) The article of claim 11, ~~15, wherein the second table is a join of the first table and another table, and~~ wherein the instructions when executed cause the system to determine the join path by further specifying a join of the second join table and a third table to produce a fourth table.

17. (Currently Amended) The article of claim 16, wherein the join query further specifies application of a second function selected from the group consisting a selection predicate applied on a complex attribute, a projection applied on a complex attribute, and a user-defined data type method, the second function being applied on a third table,

wherein the instructions when executed cause the database system to determine the join path by further applying the second function on one of the third table and ~~[[a]]~~ the fourth table with a lower cardinality[.,,]

~~wherein the fourth table is a join result of the third table and another table.~~

Appl. No. 10/056,880
Amdt. Dated July 1, 2004
Reply to Office Action of April 9, 2004

18. (Currently Amended) A database system comprising:
a storage system to store tables; and
an optimizer to receive a join query that specifies a function selected from the group consisting of a selection predicate applied on a complex attribute, a projection applied on a complex attribute, and a user-defined data type method,
the optimizer adapted to select a join plan based at least in part on a comparison of a first cost of applying the function on a first table and a second cost of applying the function on a second table.

19. (Currently Amended) The database system of claim 18, wherein the optimizer is adapted to select the join plan that applies the function on the one of the first table and second table with a lower cardinality.

20. (Original) The database system of claim 19, wherein the second table is a join result of the first table and another table.

21. (Currently Amended) The database system of claim 20, wherein the join query specifies the function being applied on the first table, and the optimizer to apply the function on the second table rather than the first table in response to determining the second cost is lower than the first cost.

22. (Original) The database system of claim 20, wherein the first and second tables are object relational tables.

23. (Currently Amended) The database system of claim 19, wherein the join query further specifies application of a second function selected from the group consisting of a selection predicate applied on a complex attribute, a projection applied on a complex attribute, and a user-defined data type method, the join query specifying the second function being applied on a third table, the optimizer adapted to select a join plan that applies the second function on one of the third table and a fourth table with a lower cardinality, the fourth table being a join result of the third table and another table.

Appl. No. 10/056,880
Amdt. Dated July 1, 2004
Reply to Office Action of April 9, 2004

24. (Original) The database system of claim 18, wherein the tables comprise object relational tables.

25. (New) The method of claim 1, wherein the second table contains a result of a join between the first table and another table,

wherein selecting the join path comprises selecting a join path in which at least one of selection and projection is applied on the second table rather than the first table, the method further comprising:

receiving a second query specifying a join of the first table and another table, the second query specifying at least one of a selection predicate applied on a non-complex attribute and a projection applied on a non-complex attribute; and

selecting another join path for the second query in which selection or projection is applied on the first table before performing a join of the first table with the another table.

26. (New) The method of claim 1, further comprising:

an optimizer module performing N-lookahead join planning in which costs for different combinations of joins of N+2 tables are determined, where N is greater than or equal to one.

27. (New) The method of claim 1, wherein the query specifies application of the function on the first table,

wherein selecting the join path comprises selecting the join path in which the function is applied on the second table, the second table containing a join result of a join of the first table and another table.

Appl. No. 10/056,880
Amdt. Dated July 1, 2004
Reply to Office Action of April 9, 2004

28. (New) The article of claim 11, wherein the instructions when executed cause the database system to:

receive a second query specifying a join of the first table and another table, the second query specifying at least one of a selection predicate applied on a non-complex attribute and a projection applied on a non-complex attribute; and

select another join path for the second query in which selection or projection is applied on one of the first and second tables before performing a join of the first table with the second table.

29. (New) The article of claim 11, wherein the instructions when executed cause the database system to:

perform N-lookahead join planning in which costs for different combinations of joins of N+2 tables are determined, where N is greater than or equal to one.

30. (New) The database system of claim 21, wherein the optimizer is to:

receive a second query specifying a join of the first table and the another table, the second query specifying at least one of a selection predicate applied on a non-complex attribute and a projection applied on a non-complex attribute,

select another join path for the second query in which selection or projection is applied on the first table before performing a join of the first table with the another table.

31. (New) The database system of claim 18, wherein the optimizer is to perform N-lookahead join planning in which costs for different combinations of joins of N+2 tables are determined, where N is greater than or equal to one.

Appl. No. 10/056,880
Amdt. Dated July 1, 2004
Reply to Office Action of April 9, 2004

32. (New) A method of performing a join in a database system, comprising:
- receiving a join query specifying a join of a first table and a second table and containing at least one of a selection predicate and a projection;
 - selecting a join path for the join query in response to determining whether the at least one of the selection predicate and projection is applied on a complex attribute,
 - wherein a first join path is selected in which the at least one of the selection predicate and projection is applied on a join table in response to determining that the at least one of the selection predicate and projection is applied on a complex attribute, the join table containing a join result of the first and second tables, and
 - wherein a second join path is selected in which the at least one of the selection predicate and projection is applied on the first table before the join in response to determining that the at least one of the selection predicate and the projection is applied on a non-complex attribute.